00202T "090TE260

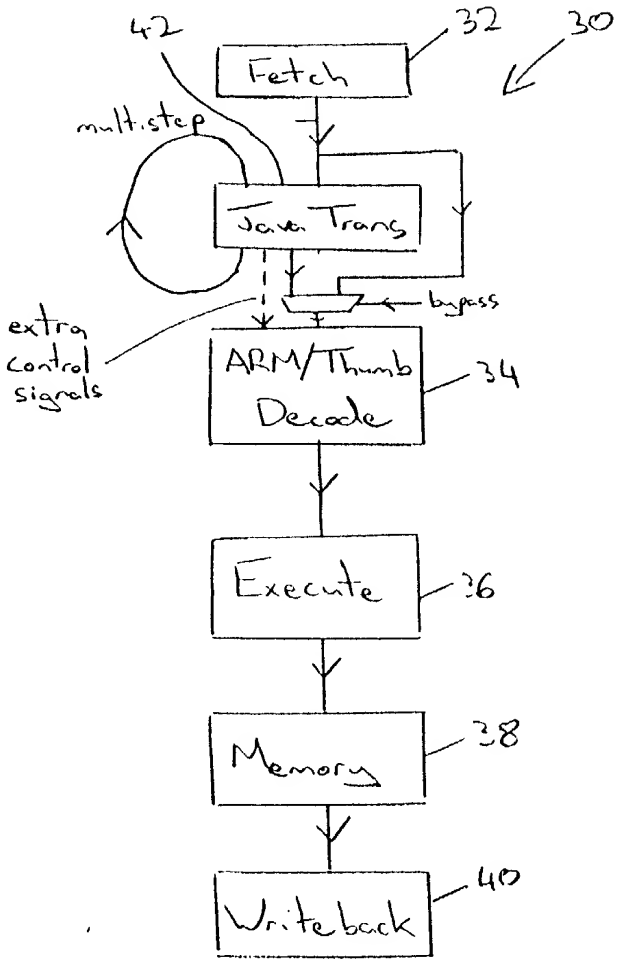

Fig. 1

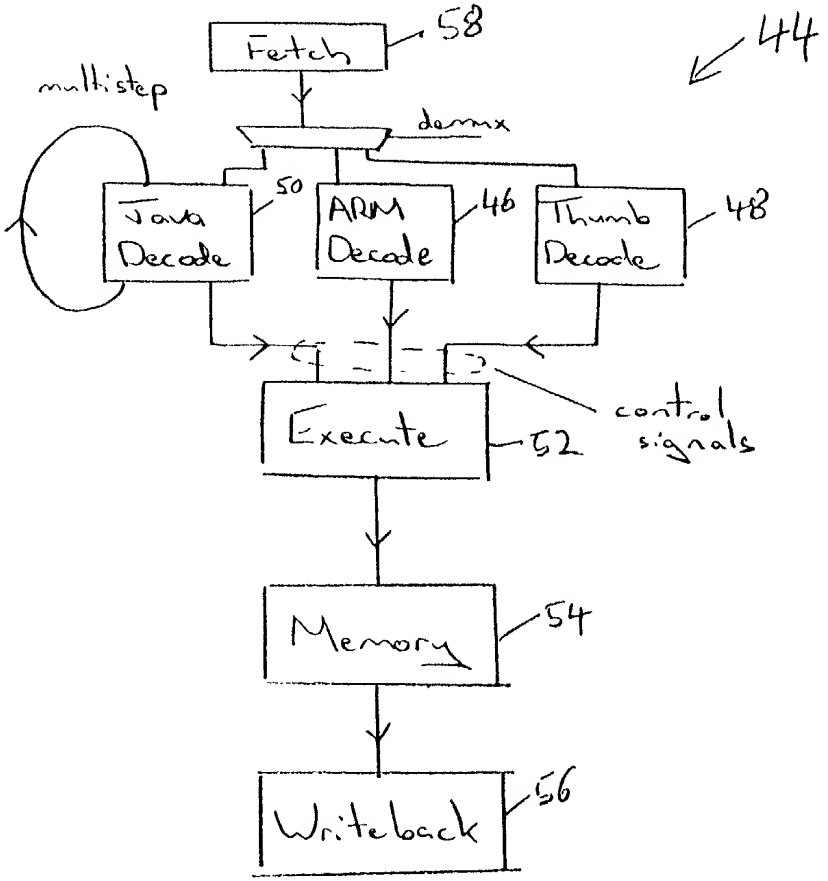

Fig. 2

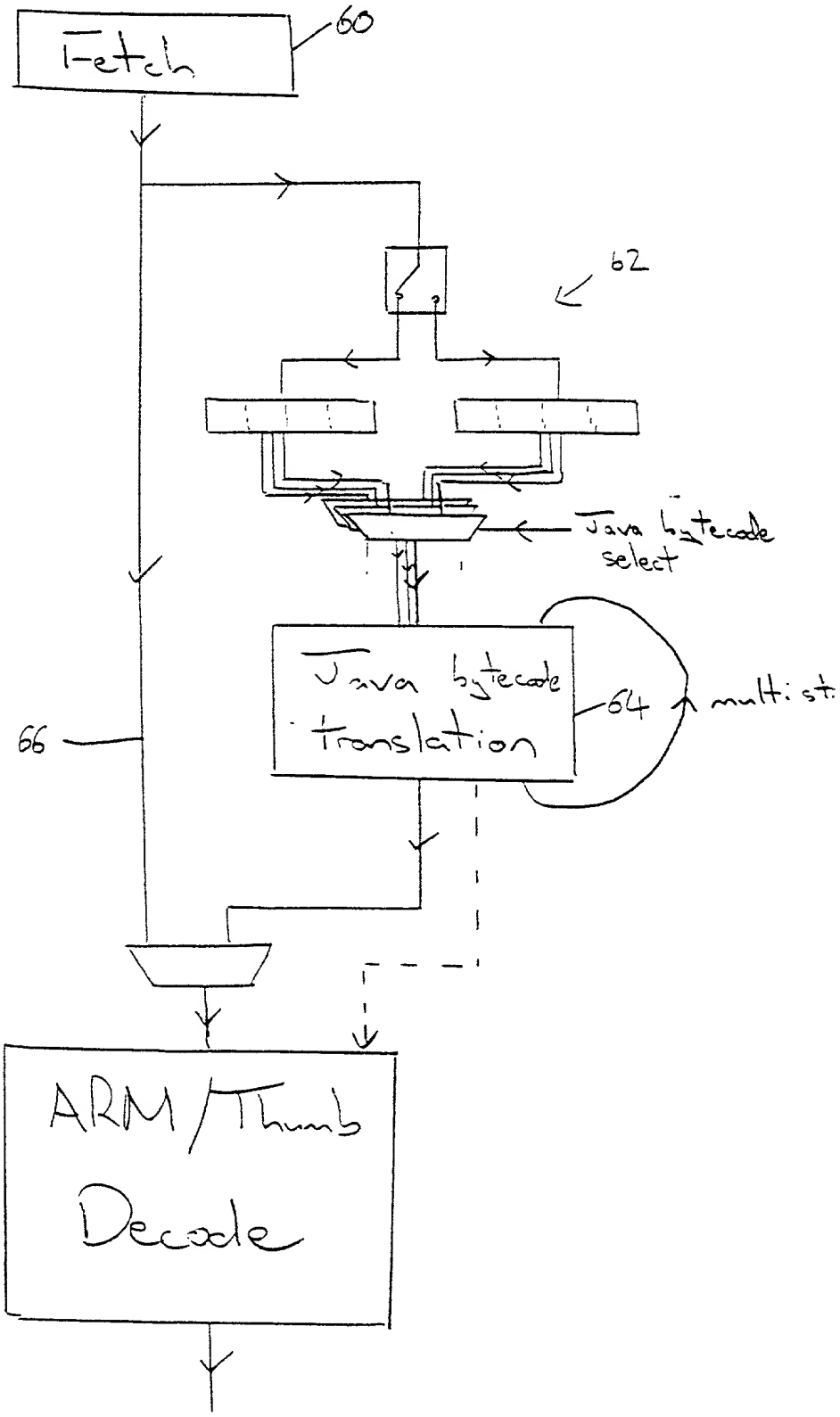
Fetch
60
62
Java bytecode select
Java bytecode translation
64
multi st
66
ARM/Thumb Decode

Fig. 3

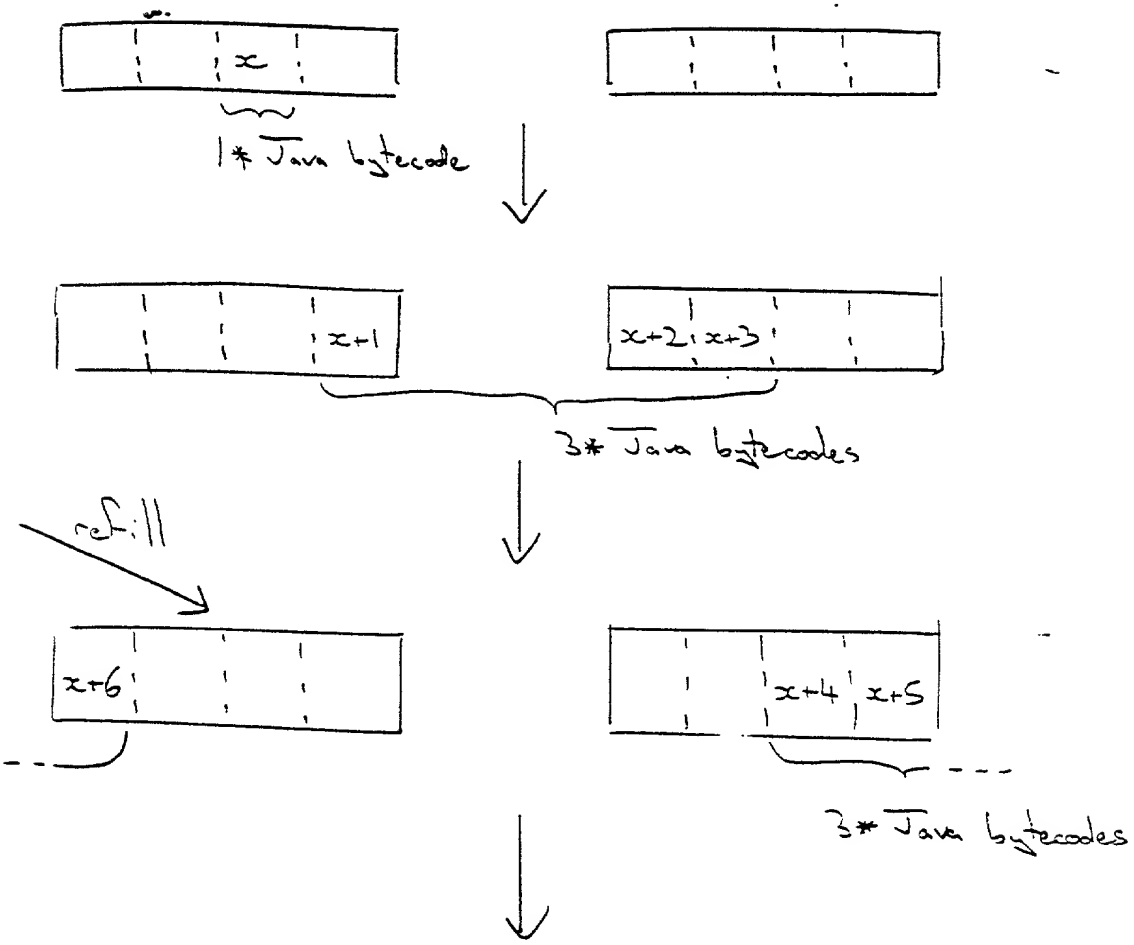
x
1* Java bytecode
x+1
x+2
x+3
3* Java bytecodes
ref:11
x+6
x+4
x+5
3* Java bytecodes

Fig. 4

0020215 090E260

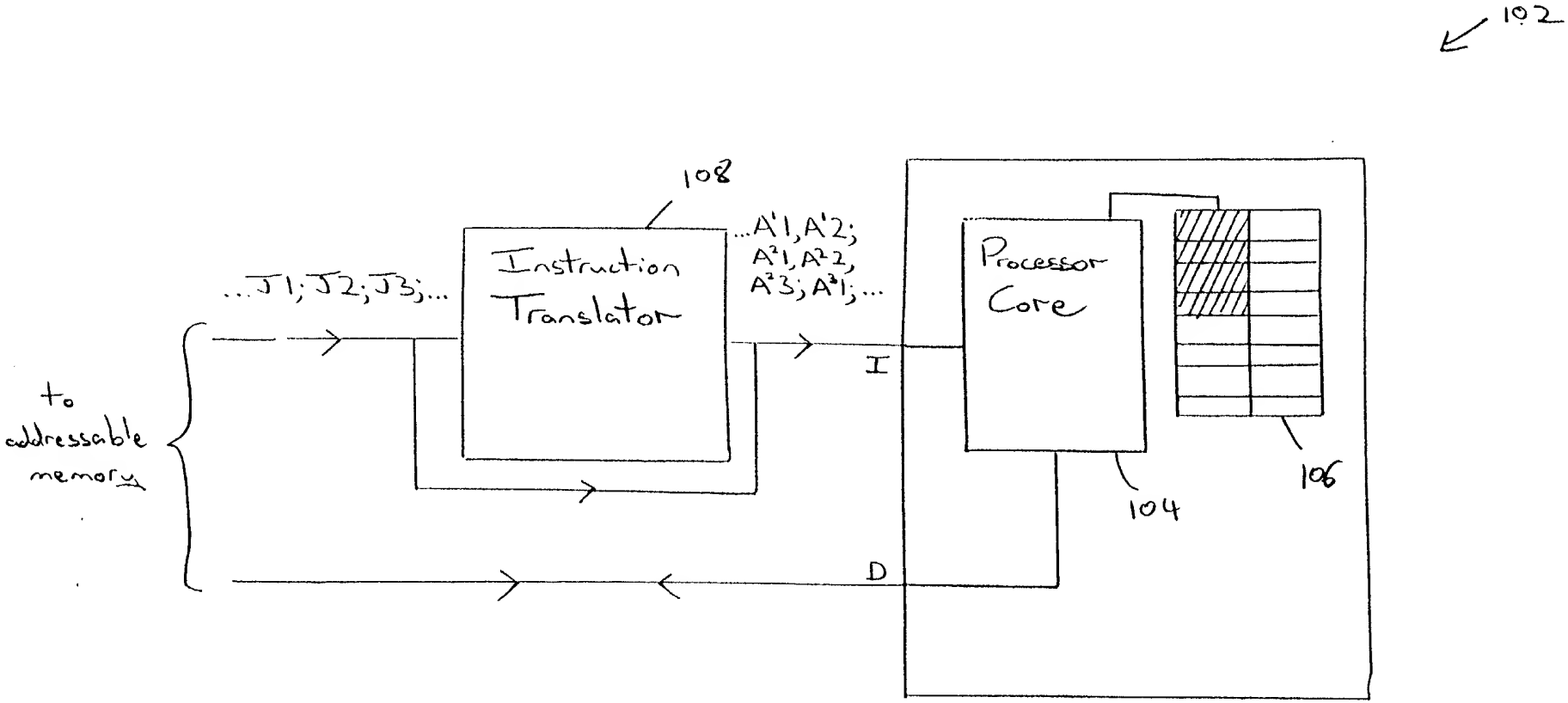

Fig. 5

-002021"090TE250

Fig. 6

Java Instruction	iadd (RF=2, RF>0)	iadd (RF=2, RF>1)	iadd (SA=-1)
ARM Instruction(s)	LDR R0,[Rstack,#4]! (POP)	LDR R3,[Rstack,#4]! (POP)	ADD R3,R3,R0
State	00000	00100	01000	00111
R0	E	SOA TOS	SOA TOS	E
R1	E	E	E	E
R2	E	E	E	E
R3	E	E	SOB TOS-1	(SOA+SOB) TOS

Java Instruction	lload' (RF=0, RE=2)	lload' (RF=0, RE>2)	lload' (RF=0, RE=2)
ARM Instructions	LDR R1,[Rvars,#4] LDR R0,[Rvars,#0]	STR R3,[RStack],#4 (PUSH)	LDR R3,[Rvars,#4] LDR R2,[Rvars,#0]
State	00111	01101	01001	10011
R0	E	SOC TOS-1	SOC TOS-1	SOC TOS-3
R1	E	SOD TOS	SOD TOS	SOD TOS-2
R2	E	E	E	SOE TOS-1
R3	(SOA+SOB) TOS	(SOA+SOB) TOS-1	E	SOF TOS

002021 "090TE260

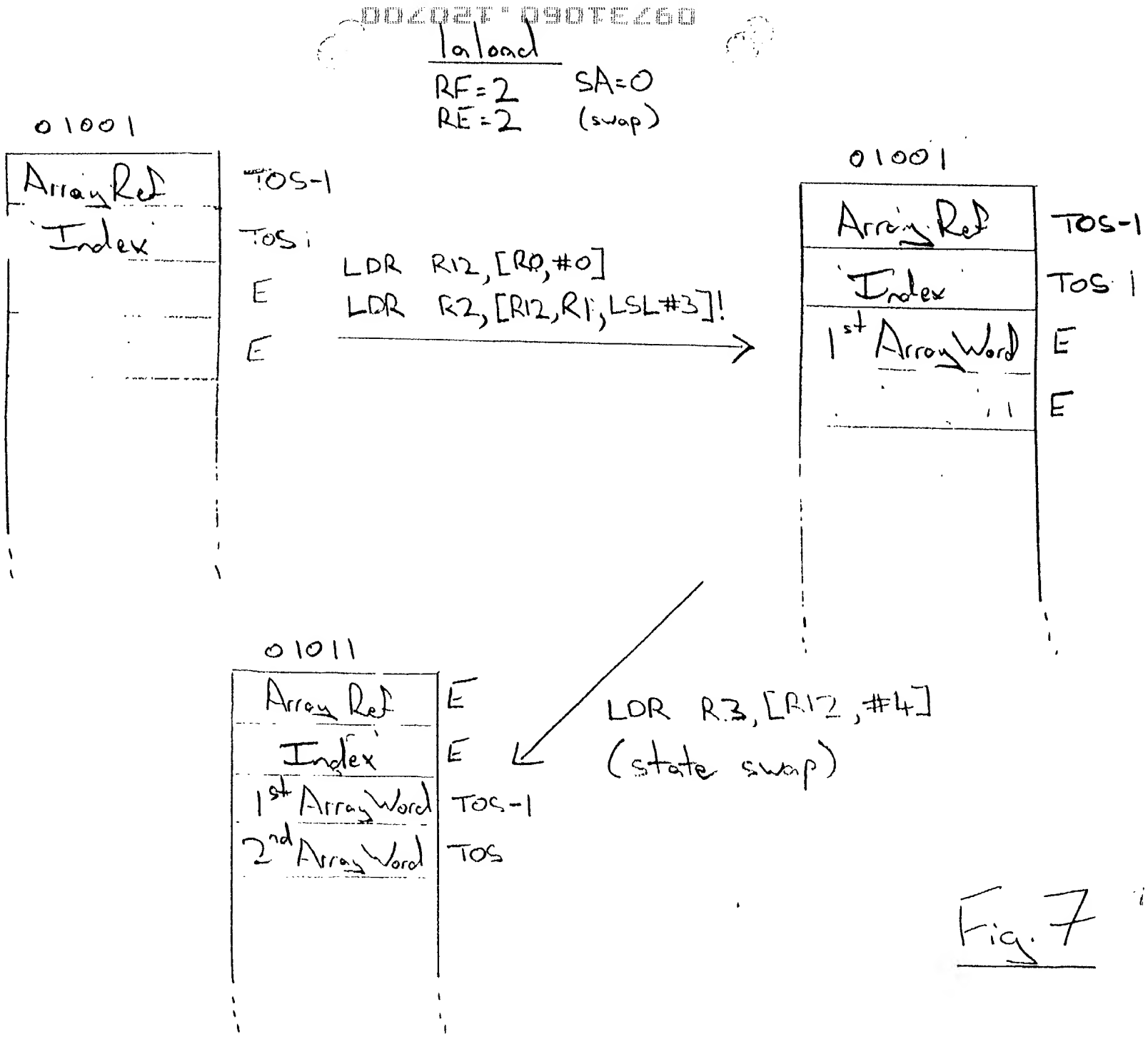

Fig. 7

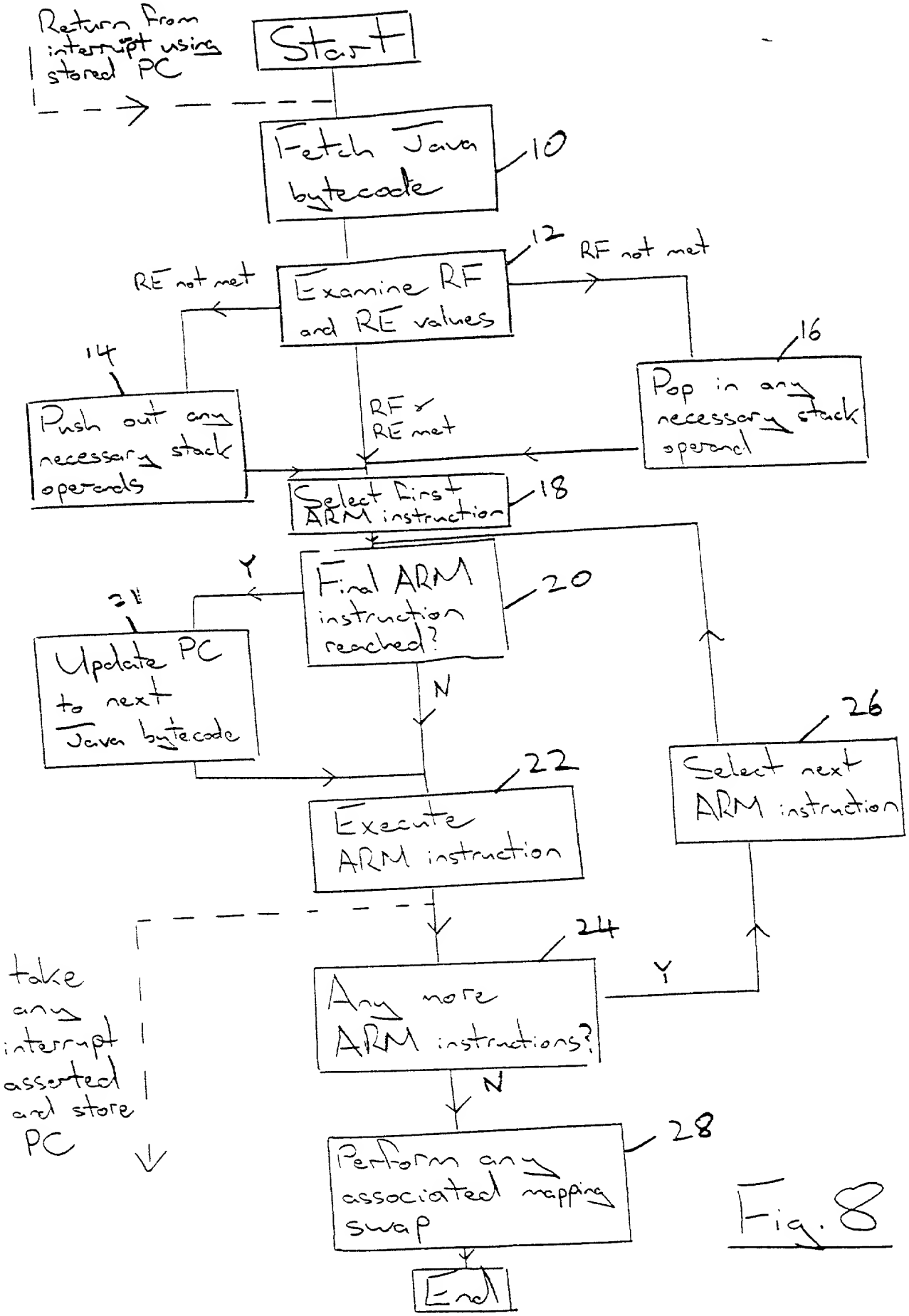
Return from interrupt using stored PC
Start
Fetch Java bytecode
10
Examine RF and RE values
12
RE not met
RF not met
14
Push out any necessary stack operands
16
Pop in any necessary stack operand
RF ✓ RE met
Select first ARM instruction
18
Final ARM instruction reached?
20
Y
N
21
Update PC to next Java bytecode
22
Execute ARM instruction
26
Select next ARM instruction
24
Any more ARM instructions?
Y
N
take any interrupt asserted and store PC
28
Perform any associated mapping swap
End

Fig. 8

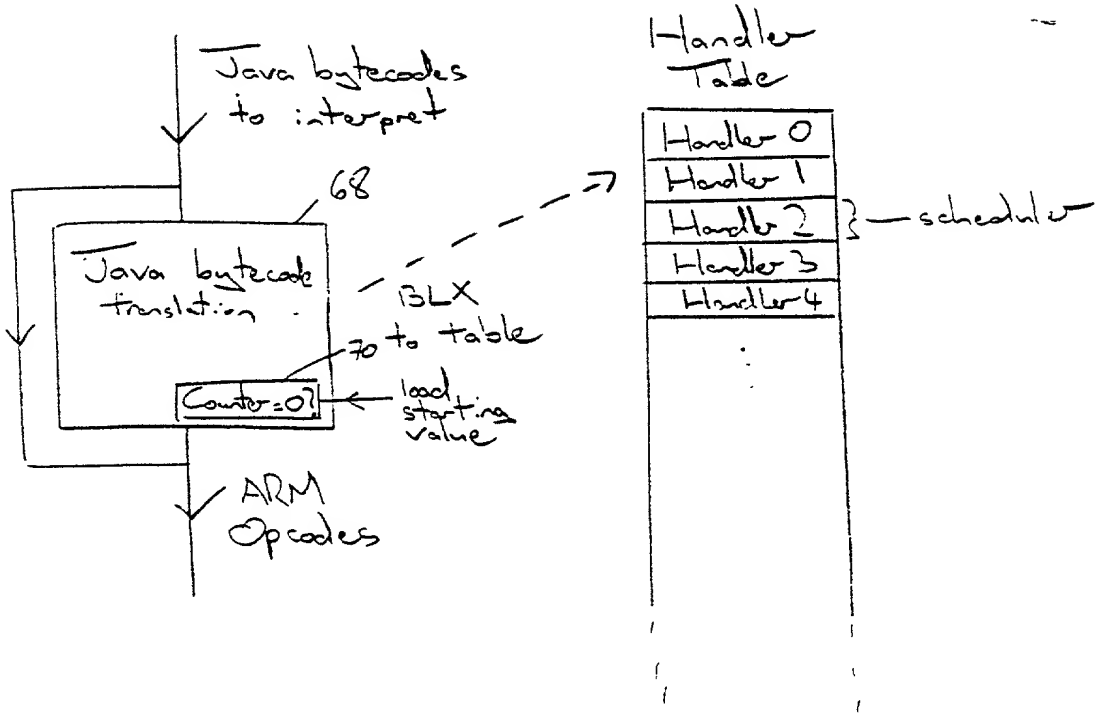
Java bytecodes to interpret
68
Java bytecode translation
BLX to table
70
Counter=0?
load starting value
ARM Opcodes
Handler Table
Handler 0
Handler 1
Handler 2
Handler 3
Handler 4
scheduler

Fig. 9

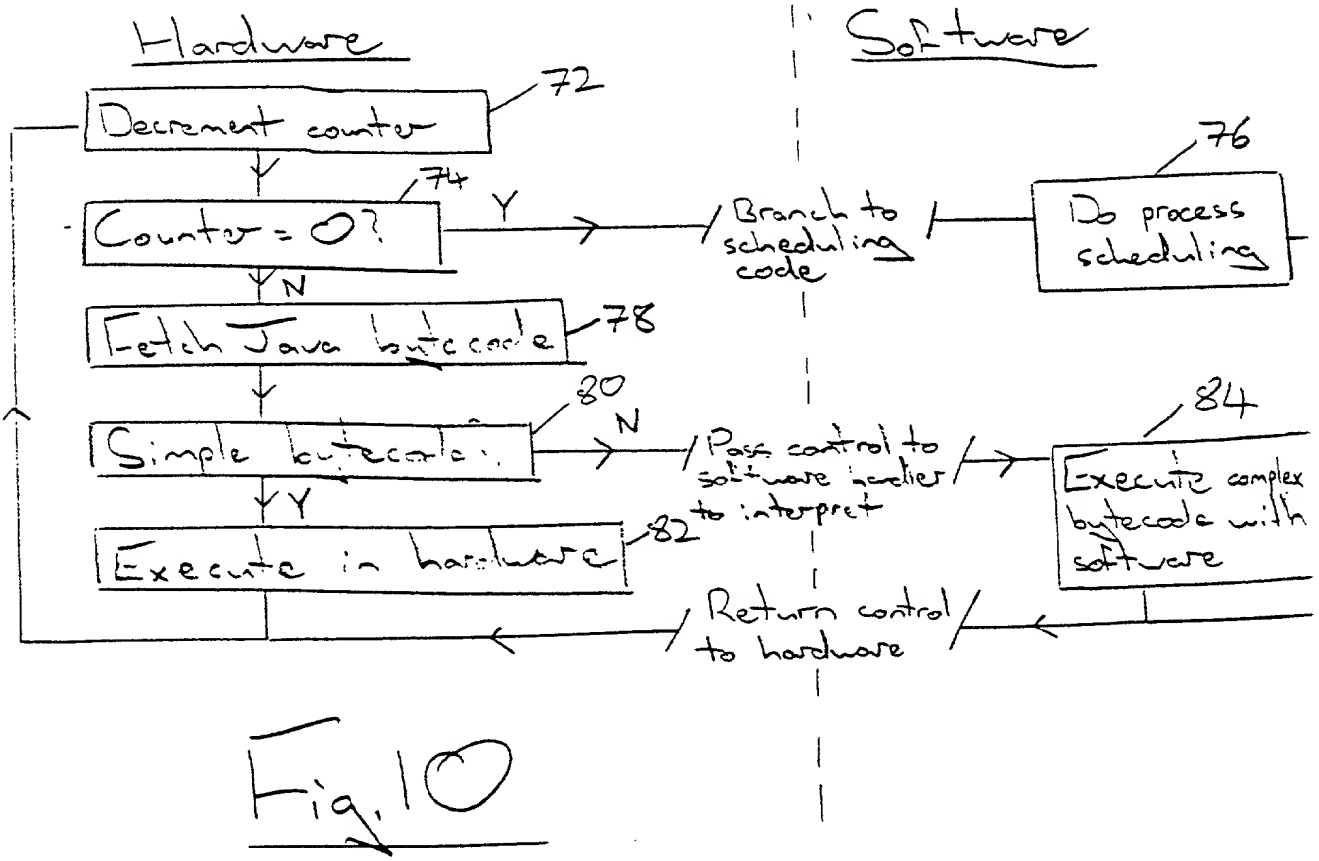
Hardware
Software
Decrement counter
72
Counter = 0?
74
Y
N
Branch to scheduling code
Do process scheduling
76
Fetch Java bytecode
78
Simple bytecode?
80
N
Y
Pass control to software handler to interpret
Execute complex bytecode with software
84
Execute in hardware
82
Return control to hardware

Fig. 10

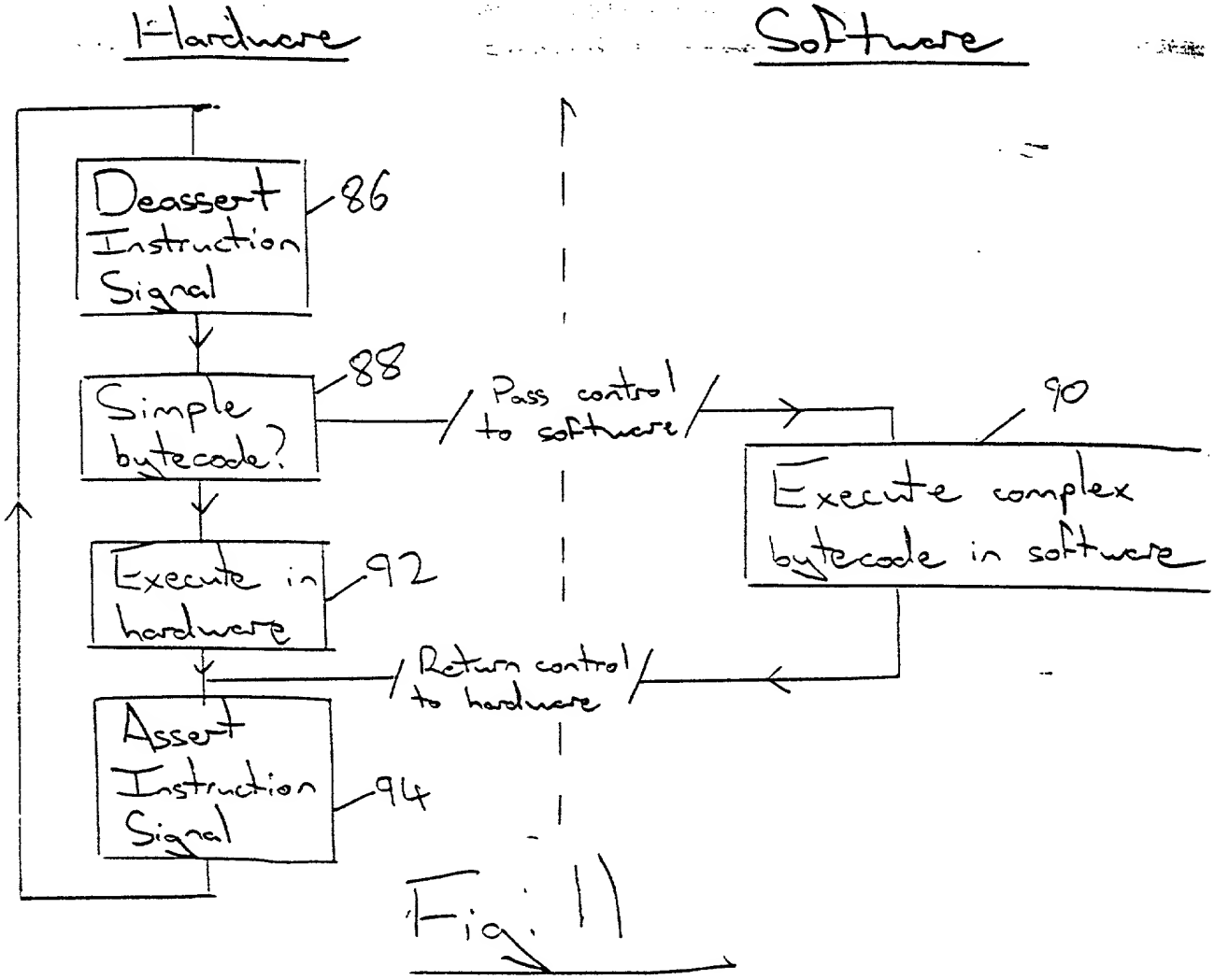
Hardware
Software
Deassert Instruction Signal
86
Simple bytecode?
88
Pass control to software
90
Execute complex bytecode in software
Execute in hardware
92
Return control to hardware
Assert Instruction Signal
94

Fig. 11

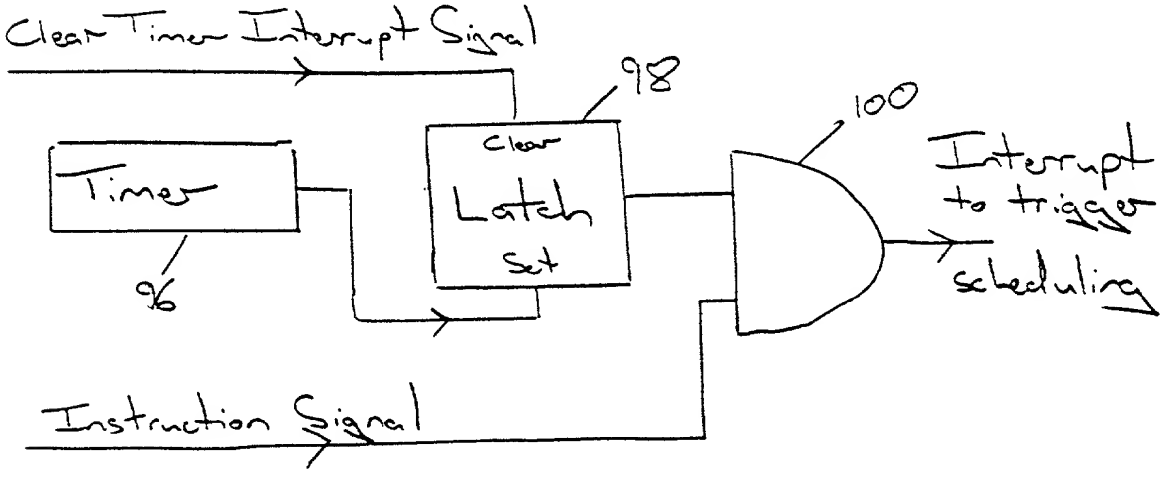
Clear Timer Interrupt Signal
98
100
Clear
Latch
Set
Timer
96
Interrupt to trigger scheduling
Instruction Signal

Fig. 12

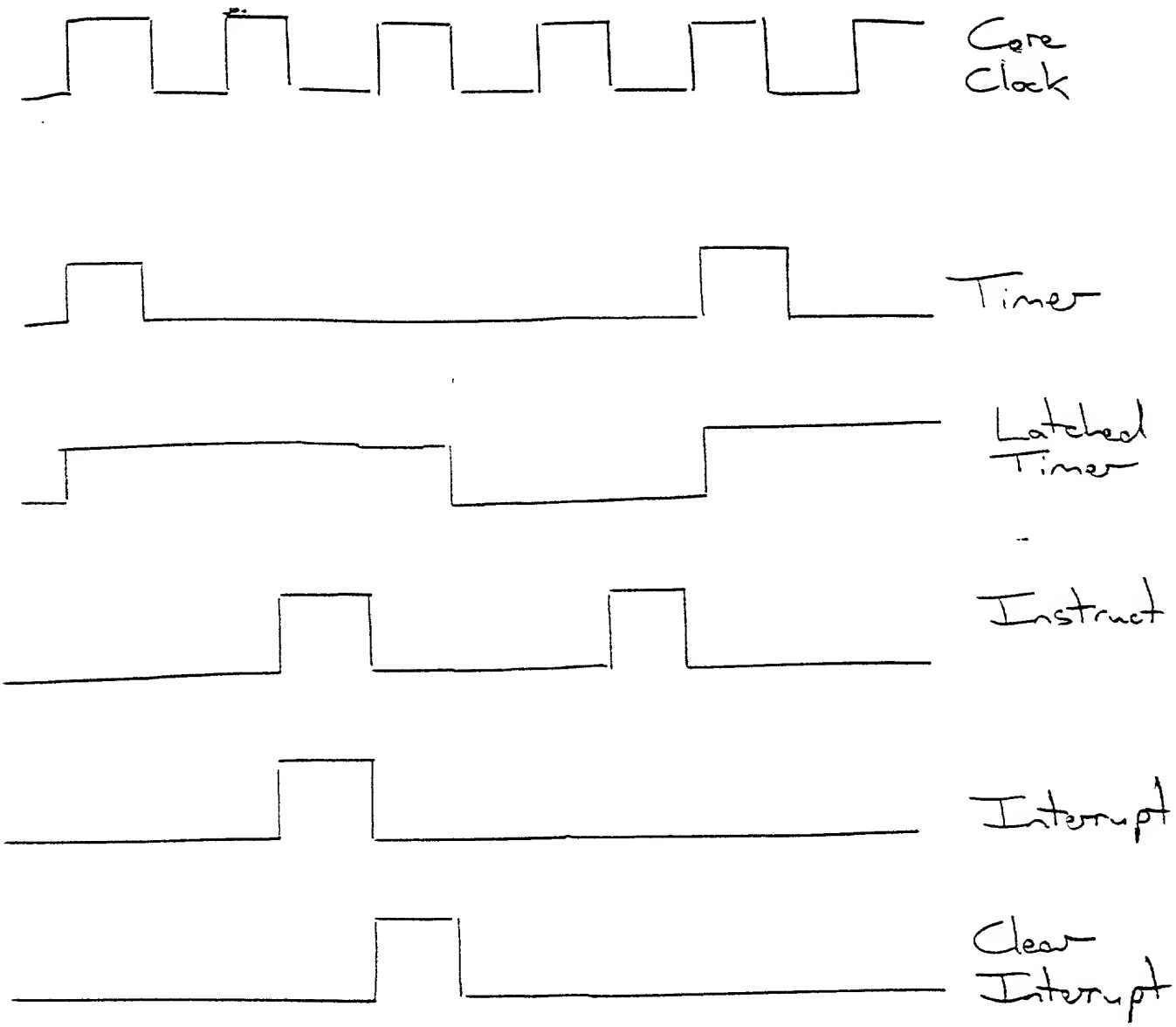
Core Clock
Timer
Latched Timer
Instruct
Interrupt
Clear Interrupt

Fig. 13